Министерство Образования, Культуры,

Исследований Республики Молдова

Технический Университет Молдовы

Департамент Программная Инженерия и Автоматика

**Отчёт**

по лабораторной работе №2

**по дисциплине «PS»**

Выполнил: ст.гр. TI-197

Шарафудинов Н.

Проверил: Romanenco A.

Кишинёв - 2022

**Лабораторная работа №2**

**Тема:** Дискретные во времени системы.

**Цель:** научиться создавать различные системы в соответствии с исходным выражением.

**Краткая теория:**

Дискретные во времени системы преобразуют входной сигнал во временную область с целью получить выходной сигнал с более желаемыми свойствами. Над входным сигналом применяются алгоритмы, состоящие из простых операций. Целью данной второй лабораторной работы является иллюстрация некоторых простых типов дискретных во времени систем, с указанием их свойств.

В линейной дискретной во времени системы для входного сигнала x[n]=αx1[n]+βx2[n] откликом будет y[n]=αy1[n]+βy2[n], где y1[n] и y2[n] отклики на последовательности x1[n] и x2[n] соответственно.

В дискретной и постоянной во времени системе откликом входного сигнала x[n]=x1[n-n0] будет являться сигнал y[n]=y1[n-n0], где n0 – целое положительное или отрицательное число и y1[n] – отклик на x1[n].

Линейная постоянная во времени система (Linear Time-Invariant – LTI) удовлетворяет обоим условиям линейности и постоянства во времени.

Если y1[n] и y2[n] являются откликами каузальной дискретной системы на входы u1[n] и u2[n] соответственно, то u1[n]= u2[n], при n<N подразумевает, что y1[n]= y2[n], при n<N.

**FIR-filter** – (Finite-duration Impulse Response или же КИХ – Конечная Импульсная Характеристика) один из видов линейных цифровых фильтров, характерной особенностью которого является ограниченность по времени его импульсной характеристики (с какого-то момента времени она становится точно равной нулю).

**IIR-filter** – (Infinite-duration Impulse Response или же БИХ – Бесконечная Импульсная Характеристика) линейный электронный фильтр, использующий один или более своих выходов в качестве входа, то есть образующий обратную связь. Основным свойством таких фильтров является то, что их импульсная переходная характеристика имеет бесконечную длину во временной области, а передаточная функция имеет дробно-рациональный вид.

**Используемые команды MATLAB**

Команды общего назначения

disp

Операторы и специальные символы

: . + - \* / ; % <

Конструкции языка

break end for if input

Элементарные матрицы и действия над ними

ones pi zeros

Элементарные функции

abs cos

Полиномиальные и интерполяционные функции

conv

Двумерная графика

axis plot stem title xlabel ylabel

Графические функции общего назначения

clf subplot

Функции работы со строками

num2str

Функции пакета обработки сигналов

filter impz

**Выполнение работы**

**% Программа P2\_1**

% Симуляция FIR фильтра произвольной длины M

% Генерация входного сигнала

clf;

n=0:150;

s1=cos(2\*pi\*0.01\*n); % Низко-частотный сигнал

s2=cos(2\*pi\*0.1\*n); % Высоко-частотный сигнал

x=s1+s2;

% Реализация FIR фильтра

M=input('Желаемая длина фильтра = ');

num=ones(1,M);

y=filter(num,1,x)/M;

% Вывод входных и выходных сигналов

subplot(2,2,1);

plot(n,s1);

axis([0, 100, -2, 2]);

xlabel('Время n'); ylabel('Амплитуда');

title('Сигнал #1');

subplot(2,2,2);

plot(n,s2);

axis([0, 100, -2, 2]);

xlabel('Время n'); ylabel('Амплитуда');

title('Сигнал #2');

subplot(2,2,3);

plot(n,x);

axis([0, 100, -2, 2]);

xlabel('Время n'); ylabel('Амплитуда');

title('Входной сигнал');

subplot(2,2,4);

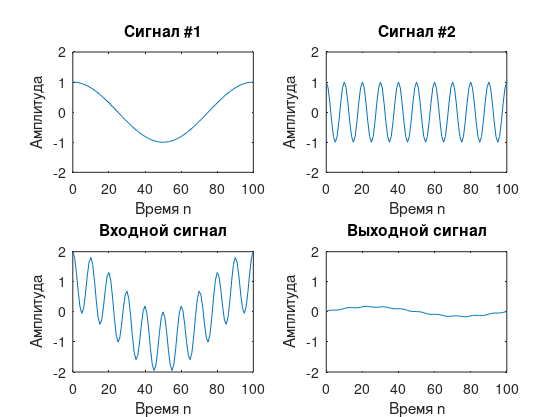
plot(n,y);

axis([0, 100, -2, 2]);

xlabel('Время n'); ylabel('Амплитуда');

title('Выходной сигнал');

axis;



**% Программа P1\_2**

% Программа P2\_2

% Генерация синусоидального входного сигнала

clf;

n=0:200;

x=cos(2\*pi\*0.05\*n);

% Вычисление выходного сигнала

x1 = [x 0 0]; % x1[n] = x[n+1]

x2 = [0 x 0]; % x2[n] = x[n]

x3 = [0 0 x]; % x3[n] = x[n-1]

y=x2.\*x2 + x1.\*x3;

y=y(2:202);

% Визуализация входного и выходного сигналов

subplot(2,1,1);

plot(n,x);

xlabel('Время n'); ylabel('Амплитуда');

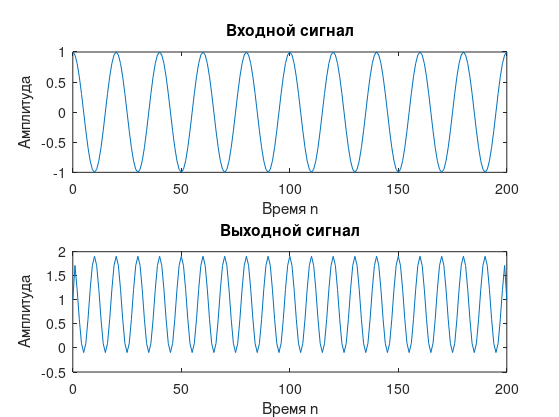
title('Входной сигнал');

subplot(2,1,2);

plot(n,y);

xlabel('Время n'); ylabel('Амплитуда');

title('Выходной сигнал');



**% Программа P1\_3**

% Программа P2\_3

% Генерация входных последовательностей

clf;

n=0:40;

a=2; b=-3;

x1=cos(2\*pi\*0.1\*n);

x2=cos(2\*pi\*0.4\*n);

x=a\*x1+b\*x2;

num=[2.2403 2.4908 2.2403];

den=[1 -0.4 0.75];

ic=[0 0]; % Установка нулевых начальных условий

y1=filter(num,den,x1,ic); % Вычисление выходного сигнала y1[n]

y2=filter(num,den,x2,ic); % Вычисление выходного сигнала y2[n]

y=filter(num,den,x,ic); % Вычисление выходного сигнала y[n]

yt=a\*y1+b\*y2;

d=y-yt; % Вычисление отклонения d[n]

% Визуализация выходных сигналов и сигнала отклонения

subplot(3,1,1);

stem(n,y);

ylabel('Амплитуда');

title('Выходной сигнал, соответствующий загруженному входному сигналу: a \cdot x\_{1}[n] + b \cdot x\_{2}[n]');

subplot(3,1,2);

stem(n,yt);

ylabel('Амплитуда');

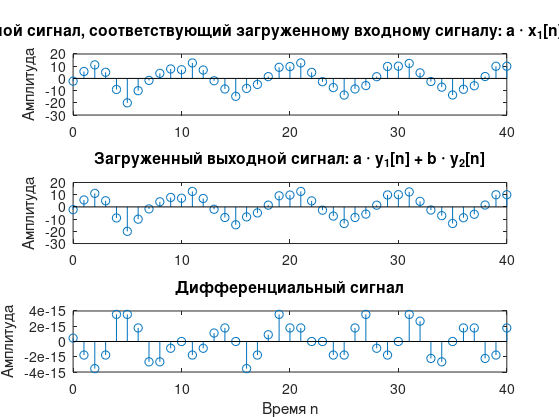
title('Загруженный выходной сигнал: a \cdot y\_{1}[n] + b \cdot y\_{2}[n]');

subplot(3,1,3);

stem(n,d);

xlabel('Время n'); ylabel('Амплитуда');

title('Дифференциальный сигнал');



**% Программа P1\_4**

% Программа P2\_4

% Генерация входных последовательностей

clf;

n=0:40; D=10; a=3.0; b=-2;

x=a\*cos(2\*pi\*0.1\*n)+b\*cos(2\*pi\*0.4\*n);

xd=[zeros(1,D) x];

num=[2.2403 2.4908 2.2403];

den=[1 -0.4 0.75];

ic=[0 0]; % Установка начальных условий

% Вычисление выходного сигнала y[n]

y=filter(num,den,x,ic);

% Вычисление выходного сигнала yd[n]

yd=filter(num,den,xd,ic);

% Вычисление сигнала разности d[n]

d=y-yd(1+D:41+D);

% Вывод графиков выходных сигналов

subplot(3,1,1);

stem(n,y);

ylabel('Амплитуда');

title('Выходной сигнал y[n]'); grid;

subplot(3,1,2);

stem(n,yd(1:41));

ylabel('Амплитуда');

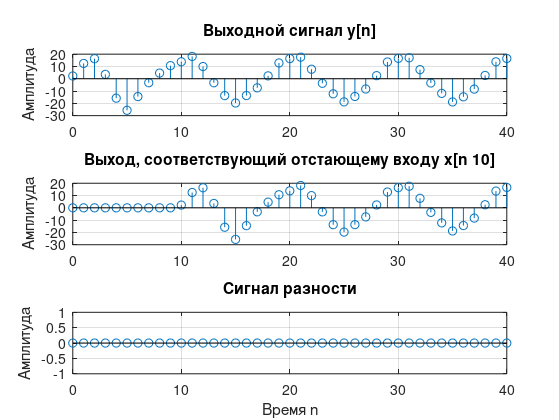
title(['Выход, соответствующий отстающему входу x[n ',num2str(D),']']); grid;

subplot(3,1,3);

stem(n,d);

xlabel('Время n'); ylabel('Амплитуда');

title('Сигнал разности'); grid;



**% Программа P1\_5**

% Программа P2\_5

% Вычисление импульсного отклика y

clf;

N=40;

num=[2.2403 2.4908 2.2403];

den=[1 -0.4 0.75];

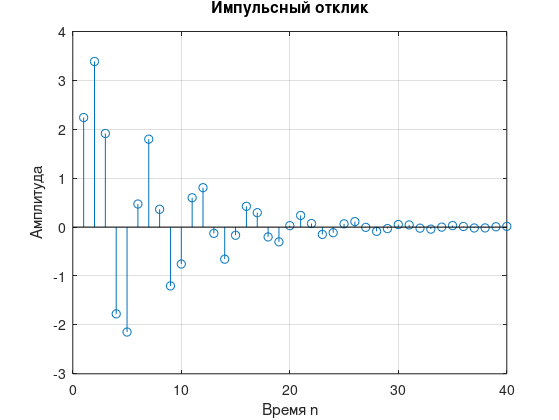
y=impz(num,den,N);

% Графическое отображение импульсного отклика

stem(y);

xlabel('Время n'); ylabel('Амплитуда');

title('Импульсный отклик'); grid;



**% Программа P2\_6**

% Реализация каскада

clf;

x=[1 zeros(1,40)]; % Генерация входного сигнала

n=0:40;

% Коэффициенты системы четвёртого порядка

den=[1 1.6 2.28 1.325 0.68];

num=[0.06 -0.19 0.27 -0.26 0.12];

% Вычисление выходного сигнала системы четвёртого порядка

y=filter(num,den,x);

% Коэффициенты систем второго порядка

num1=[0.3 -0.2 0.4]; den1=[1 0.9 0.8];

num2=[0.2 -0.5 0.3]; den2=[1 0.7 0.85];

% Выходной сигнал y1[n] первой стадии каскада

y1=filter(num1,den1,x);

% Выходной сигнал y2[n] второй стадии каскада

y2=filter(num2,den2,y1);

% Смещение между y[n] и y2[n]

d=y-y2;

% Графики выходных сигналов и смещения

subplot(3,1,1);

stem(n,y);

ylabel('Амплитуда');

title('Вывод реализации четвёртого порядка'); grid;

subplot(3,1,2);

stem(n,y2);

ylabel('Амплитуда');

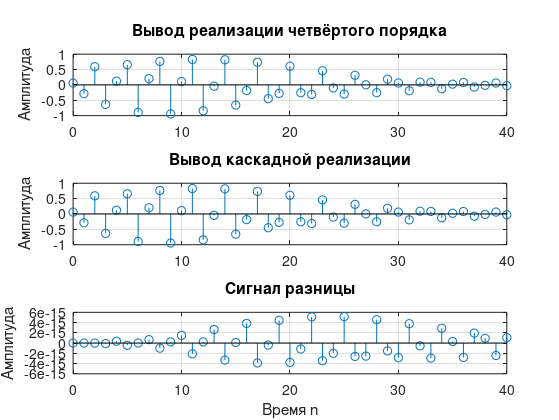
title('Вывод каскадной реализации'); grid;

subplot(3,1,3);

stem(n,d);

xlabel('Время n'); ylabel('Амплитуда');

title('Сигнал разницы'); grid;



**% Программа P2\_7**

clf;

h=[3 2 1 -2 1 0 -4 0 3]; % импульсный отклик

x=[1 -2 3 -4 3 2 1]; % входная последовательность

y=conv(h,x);

n=0:14;

subplot(2,1,1);

stem(n,y);

xlabel('Время n'); ylabel('Амплитуда');

title('Выходной сигнал, полученный свёрткой'); grid;

x1=[x zeros(1,8)];

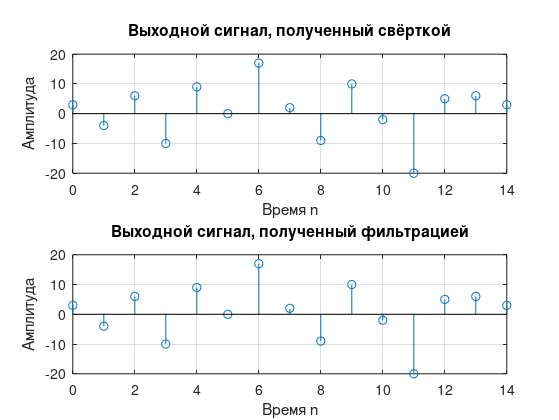
y1=filter(h,1,x1);

subplot(2,1,2);

stem(n,y1);

xlabel('Время n'); ylabel('Амплитуда');

title('Выходной сигнал, полученный фильтрацией'); grid;



**% Программа P2\_8**

% Тест на стабильность, основанный на сумме абсолютных

% значений элементов последовательности импульсного отклика

clf;

num=[1 -0.8]; den=[1 1.5 0.9];

N=200;

h=impz(num,den,N+1);

parsum=0;

for k=1:N+1;

parsum=parsum+abs(h(k));

if abs(h(k))<10^(-6), break, end

end

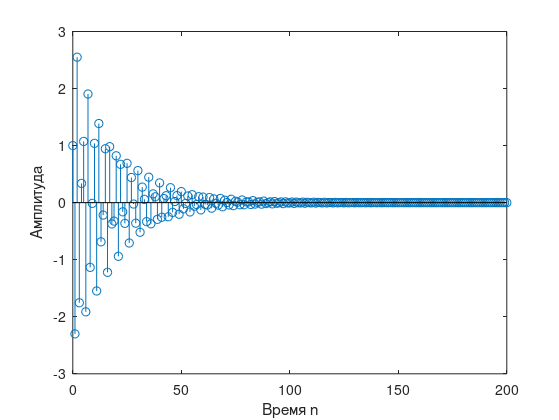
% Вывод импульсного отклика

n=0:N;

stem(n,h);

xlabel('Время n'); ylabel('Амплитуда');

disp('Значение='); disp(abs(h(k))); % Печать значений abs(h(k))



**% Программа P2\_9**

% Генерация входного сигнала

clf;

n=0:299;

x1=cos(2\*pi\*10\*n/256);

x2=cos(2\*pi\*100\*n/256);

x=x1+x2;

% Вычисление выходных последовательностей

num1=[0.5 0.27 0.77];

y1=filter(num1,1,x); % Выход системы #1

den2=[1 -0.53 0.46];

num2=[0.45 0.5 0.45];

y2=filter(num2,den2,x); % Выход системы #2

% Графики выходных последовательностей

subplot(2,1,1);

plot(n,y1); axis([0 300 -2 2]);

ylabel('Амплитуда');

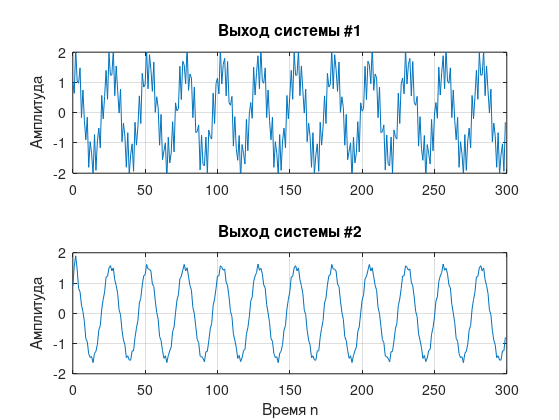
title('Выход системы #1'); grid;

subplot(2,1,2);

plot(n,y2); axis([0 300 -2 2]);

xlabel('Время n'); ylabel('Амплитуда');

title('Выход системы #2'); grid;



**Вывод:**

В данной лабораторной работе были рассмотрены дискретные во времени системы.Также была рассмотрена работа с функциями matlab для построения графиков: axis, grid, legend, plot, stairs, stem, title, xlabel, ylabel, clf, subplot и т.д., Помимо этого, были описаны **FIR-filter** и **IIR-filter**.